

# A Contribution to the Numerics of Polynomials and Matrix Polynomials

Sigurd Falk <sup>†</sup>

<sup>†</sup>Leibniz Universität Hannover, Germany  
Institute of Theoretical Electrical Engineering  
Email: falk@tet.uni-hannover.de

**Abstract**— In this paper some algorithms will be presented which can be used for the calculation of zeros of polynomials and eigenvalues of polynomial matrices with a multiplicity larger than one. The numerical values calculated with MATLAB are used as starting values. The reliability of the algorithms is demonstrated by means of 8 examples.

## 1. Formulation of the Problem

Let be a matrix eigenvalue equation

$$\mathbf{y}^T \mathbf{F}(\lambda) = \mathbf{0}^T, \quad \mathbf{F}(\lambda) \mathbf{x} = \mathbf{0} \quad (1)$$

with a polynomial matrix of the order  $n$  and the degree  $\rho$

$$\mathbf{F}(\lambda) = \mathbf{A}_0 + \mathbf{A}_1 \lambda + \mathbf{A}_2 \lambda^2 + \cdots + \mathbf{A}_\rho \lambda^\rho; \quad \det \mathbf{A}_\rho \neq 0 \quad (2)$$

and complex-valued coefficient matrices  $\mathbf{A}_0, \dots, \mathbf{A}_\rho$ .

In the following the eigenvalues of  $\mathbf{F}(\lambda)$

$$\lambda_1, \lambda_2, \dots, \lambda_m; \quad m = \rho \cdot n \quad (3)$$

defined as zeros of the characteristic polynomial

$$\det \mathbf{F}(\lambda) = f(\lambda) = a_0 + a_1 \lambda + a_2 \lambda^2 + \cdots + a_m \lambda^m \quad (4)$$

will be calculated by means of a Padé function<sup>1</sup>

$$p(\lambda) = \frac{f(\lambda)}{z(\lambda)}, \quad (5)$$

where  $z(\lambda)$  is a polynomial of degree  $\leq m$ .

Choosing  $z(\lambda) = -f'(\lambda)$  the Padé function

$$p(\lambda) = \frac{f(\lambda)}{-f'(\lambda)} \quad (6)$$

with an interesting property is obtained.

If the polynomial  $f(\lambda)$  possesses a zero  $a$  with the multiplicity  $\nu$  then it can be represented by

$$f(\lambda) = (\lambda - a)^\nu \cdot z(\lambda), \quad z(a) \neq 0. \quad (7)$$

It follows

$$f'(\lambda) = \nu(\lambda - a)^{\nu-1} \cdot z(\lambda) + (\lambda - a)^\nu \cdot z'(\lambda), \quad (8)$$

or

$$f'(\lambda) = (\lambda - a)^{\nu-1} \left[ \underbrace{\nu \cdot z(\lambda) + (\lambda - a) \cdot z'(\lambda)}_{\varphi(\lambda)} \right], \quad (9)$$

where

$$\varphi(a) = \nu \cdot z(a) + 0 \neq 0. \quad (10)$$

Therefore, the following theorem can be formulated:

The Padé function (6) possesses only zeros with the multiplicity  $\nu = 1$ .

<sup>1</sup>Henri Eugene Padé, French mathematician, 1863-1953

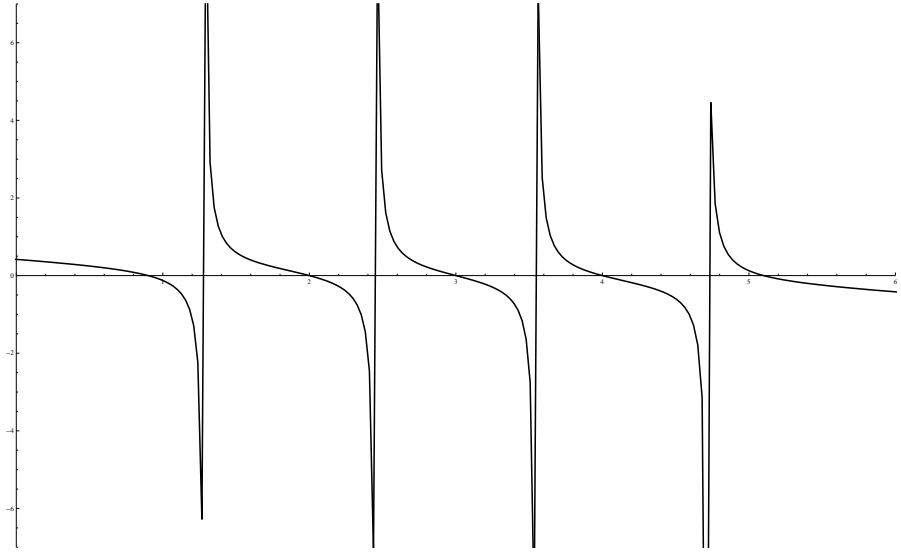


Figure 1: Typical graph of a Padé function

## 2. Algorithms

For users only such algorithms are of interest which calculate zeros also if their multiplicity is larger than one. Therefore, Newton's tangent method has to be excluded. In numerical applications there are three methods that can be used in a successful manner.

2a) A method that is based on the Padé function (6) leads to

$$\Lambda_{j+1} = \Lambda_j + p(\Lambda_j); \quad j = 1, 2, \dots \quad (11)$$

2b) A method that is founded on Halley's function

$$h(\lambda) = \frac{p(\lambda)}{1 + p(\lambda) \cdot q(\lambda)} \quad (12)$$

where

$$q(\lambda) = \frac{f''(\lambda)}{f'(\lambda)}. \quad (13)$$

leads to

$$\Lambda_{j+1} = \Lambda_j + h(\Lambda_j); \quad j = 1, 2, \dots \quad (14)$$

2c) A method that is based on the test polynomials

$$f_k(\lambda) = a_0 + (-1)^k \cdot a_2 \lambda^2 + (-2)^k \cdot a_3 \lambda^3 + \dots + (1 - m)^k \cdot a_m \lambda^m; \quad (15)$$

$k = 1, 2, \dots, \nu$ , where  $\nu$  is the multiplicity of the zero under consideration.

If the prescribed polynomial  $f_0(\lambda)$  possesses a zero  $\tilde{\lambda}$  with the multiplicity  $\nu$  then each of the Padé functions

$$P_1(\lambda) = \frac{f_0(\lambda)}{f_1(\lambda)}, \quad P_2(\lambda) = \frac{f_1(\lambda)}{f_2(\lambda)}, \dots, \quad P_\nu(\lambda) = \frac{f_{\nu-1}(\lambda)}{f_\nu(\lambda)} \quad (16)$$

possesses this zero with the multiplicity one. With

$$p_\nu(\lambda) = P_\nu(\lambda) \cdot \lambda \quad (17)$$

the algorithm can be formulated by

$$\Lambda_{j+1} = \Lambda_j + p_\nu(\Lambda_j); \quad j = 1, 2, \dots \quad (18)$$

or

$$\Lambda_{j+1} = [1 + P_v(\Lambda_j)] \cdot \Lambda_j; \quad j = 1, 2, \dots \quad (19)$$

All three algorithms convergence quadratic if the starting value is chosen in a suitable interval that includes the desired zero. This condition is fulfilled if MATLAB results are used as starting values.

### 3. Exploration

In order to obtain suitable approximated values for the start of the algorithms an exploration is needed where three cases 3a1), 3a2) and 3b) have to be distinguish.

3a) The coefficients of the prescribed polynomial (4) as well as the zeros are real. Then, the Padé function (6) is used.

3 a1) The usual regula falsi method.

If for two arbitrary test points  $\lambda_1$  and  $\lambda_2$  a change of sign occur with

$$\lambda_1 < \lambda_2; \quad p(\lambda_1) > 0, p(\lambda_2) < 0, \quad (20)$$

then a zero of the function  $p(\lambda)$  is placed between  $\lambda_1$  and  $\lambda_2$  and therefore also a zero of  $f(\lambda)$  exists possibly with a multiplicity larger than one.

Using the regula falsi method

$$\lambda_3 = \lambda_1 - \frac{p(\lambda_1)}{\Delta_2} \quad (21)$$

with the difference quotient

$$\Delta_2 = \frac{p(\lambda_2) - p(\lambda_1)}{\lambda_2 - \lambda_1} \quad (22)$$

a first approximated value (in general crude) for a zero is received.

3 a2) Regula falsi method with acceleration

The approximated value  $\lambda_3$  can be improved in the following manner. If a further difference quotient

$$\Delta_3 = \frac{p(\lambda_3) - p(\lambda_1)}{\lambda_3 - \lambda_1} \quad (23)$$

and the terms

$$Q_2 = \frac{p(\lambda_2)}{p(\lambda_1)}, \quad Q_3 = \frac{p(\lambda_3)}{p(\lambda_1)} \quad (24)$$

are defined an improved approximated value  $\lambda_4$  is calculated by

$$\lambda_4 = \lambda_1 - \frac{p(\lambda_2) - p(\lambda_3)}{Q_2\Delta_3 - Q_3\Delta_2} \quad (25)$$

and this scheme can be continued as follows

First Step.

Replace in (21) to (25) the indexes 1,2 and 3 through 2,3 and 4 and calculate  $\lambda_5$ ;  $p(\lambda_5)$ .

Second Step.

Replace in (21) to (25) the indexes 1,2 and 3 through 3,4 and 5 and calculate  $\lambda_6$ ;  $p(\lambda_6)$ .

This iteration process has to be broken if

a) the condition

$$|p(\lambda_\mu)| \leq 10^{-\sigma} \quad (26)$$

is fulfilled or

b) stop the iteration at a certain iteration step  $\mu$  without considering a stopping criteria.

Now, we discuss the exploration process. After choosing a step-size  $\delta$  we calculate on the  $\lambda$ -axis pairs of values

$$\lambda_j; p(\lambda_j); \quad j = 1, 2, \dots \quad (27)$$

beginning from zero until a first, second, third, etc. change of sign is found.

3 b) In order to calculate also the negative zeros the co-function

$$\hat{p}(\lambda) = -\frac{f(-\lambda)}{-f'(-\lambda)} \quad (28)$$

the sequence of steps (23). (24) and (25) have to be carried out until all  $m$  zeros are calculated.

3c) Much more tedious is the exploration within the complex plane since no change of sign in the sense of (20) is available.

3d) Diagonal dominant polynomial matrices

In the case of distinct diagonal dominance of a matrix the  $m = \rho \cdot n$  zeros of the equations

$$f_{jj}(\lambda) = 0; \quad j = 1, 2, \dots, n \quad (29)$$

are suitable starting points for the in section 2 presented algorithms. In the case of  $\rho = 2$  we have to solve  $n$  quadratic equations; see also the second example.

#### 4. Eigenvalues of a Polynomial Matrix

In the following we consider equation (1)

$$\mathbf{F}(\lambda)\mathbf{x} = \mathbf{0} \quad (30)$$

with the polynomial matrix (2).

Let be  $\lambda_k$  an eigenvalue with the multiplicity one, then the matrix

$$\mathbf{F}(\lambda_k) \quad (31)$$

has the rank  $n - 1$ .

a) Transformation of Gauß

If necessary a column pivot search as well as the changing of two rows will be arranged such that the matrix (31) has the form

$$\hat{\mathbf{F}}(\lambda_k) = \begin{pmatrix} \hat{\mathbf{N}}_k & \mathbf{w}_k \\ \mathbf{0}^T & 0 \end{pmatrix} \quad (32)$$

where  $\hat{\mathbf{N}}_k$  is regular upper rectangular matrix of the order  $n - 1$  and the column  $\mathbf{w}_k$  has the length  $n - 1$ .

b) Transformation due to Jordan in the order<sup>2</sup>

$$n - 1, n - 2, \dots, 2. \quad (33)$$

Therefore,

$$\hat{\mathbf{F}}(\lambda_k) = \begin{pmatrix} \mathbf{D}_k & \mathbf{z}_k \\ \mathbf{0}^T & 0 \end{pmatrix} \quad (34)$$

with a regular diagonal matrix  $\mathbf{D}_k$  of the order  $n - 1$ .

It is easy to see that the desired eigenvector is

$$\mathbf{x}_k = \begin{pmatrix} \mathbf{D}_k^{-1} \mathbf{z}_k \\ -1 \end{pmatrix}. \quad (35)$$

---

<sup>2</sup>Wilhelm Jordan, Geometer, 1842-1899

Moreover, since the system of equations (1) is homogeneous,

$$\hat{\mathbf{x}}_k = \alpha_k \cdot \mathbf{x}_k, \quad \alpha_k \neq 0 \quad (36)$$

is also an eigenvector. The factor  $\alpha_k$  can be determined such that  $\hat{\mathbf{x}}_k$  is orthonormal

$$\hat{\mathbf{x}}_k^* \hat{\mathbf{x}}_k = 1, \quad (37)$$

but we choose

$$\alpha_k = 1. \quad (38)$$

### Multiple Eigenvalues

Let be  $\lambda_k$  an eigenvalue with the multiplicity  $\nu_k$  and  $r_k$  the rank deficiency of the matrix  $\mathbf{F}(\lambda_k)$ , where we have

$$r_k \leq \nu_k. \quad (39)$$

Now, the matrix (32) has the form

$$\tilde{\mathbf{F}}(\lambda_k) = \begin{pmatrix} \tilde{\nabla} & \mathbf{0} \\ \mathbf{0}^T & \mathbf{0} \end{pmatrix} \quad (40)$$

where the zero matrix in the right lower corner has the order  $r_k$ .

Corresponding to (32) to (35) we have

$$\tilde{\mathbf{F}}(\lambda_k) = \begin{pmatrix} \mathbf{D}_k & \tilde{\mathbf{Z}}_k \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \quad (41)$$

and therefore

$$\mathbf{X}_k = \begin{pmatrix} \mathbf{D}_k^{-1} \tilde{\mathbf{Z}}_k \\ -\mathbf{I}_{r_k} \end{pmatrix} = (\mathbf{x}_1 \ \mathbf{x}_2 \ \cdots \ \mathbf{x}_{r_k}). \quad (42)$$

The vectors  $\mathbf{x}_i$  ( $i = 1, \dots, r_k$ ) are the  $r_k$  linear independent eigenvectors of  $\lambda_k$  that can be normed with respect to (36) - (38).

If we have  $r_k < \nu_k$  then the  $r_k$  eigenvectors (42) can be complemented by generalized eigenvectors; cf. [10].

Now, we consider the left eigenvectors

$$\mathbf{y}^T \mathbf{F}(\lambda) = \mathbf{0} \quad (43)$$

where after a transposition of this equation it follows

$$[\mathbf{y}^T \mathbf{F}(\lambda)]^T = [\mathbf{0}^T]^T \Rightarrow \mathbf{F}(\lambda)^T \mathbf{y} = \mathbf{0} \quad (44)$$

If  $\mathbf{F}$  is replaced by  $\mathbf{F}^T$  the concepts of this section can be used.

## 5. The EPC-Transformation

The algorithm described in [10] based on the allocation of  $m$  pairwise different interpolation values

$$\sigma_1, \sigma_2, \dots, \sigma_m, \quad (45)$$

which have to be chosen in suitable manner. With these values the following interpolation polynomials are defined

$$g_k(\lambda) = \prod_{\substack{j=1 \\ j \neq k}}^m (\sigma_j - \lambda); \quad k = 1, 2, \dots, m \quad (46)$$

and therefore the Padé functions

$$P_k(\lambda) = \frac{f(\lambda)}{g_k(\lambda)} \cdot \frac{1}{a_m}; \quad k = 1, 2, \dots, m. \quad (47)$$

For  $\lambda = \sigma_k$  we obtain the defects (as denoted in [1] and [2])

$$d_k = \frac{f(\sigma_k)}{g_k(\sigma_k)} \cdot \frac{1}{a_m}; \quad k = 1, 2, \dots, m. \quad (48)$$

and the corresponding so-called main values

$$H_k = \sigma_k - d_k; \quad k = 1, 2, \dots, m. \quad (49)$$

These values will be collected in the following list

$$L_m = \left( \begin{array}{ccc} \text{Interpolation Values} & \text{Defects} & \text{Main Values} \\ \hline \sigma_1 & d_1 & H_1 \\ \sigma_1 & d_1 & H_1 \\ \vdots & \vdots & \vdots \\ \sigma_m & d_m & H_m \end{array} \right), \quad (50)$$

that includes the entire information of the polynomial matrix (2). The order of the rows is arbitrary. The control equation

$$\sum_{j=1}^m H_j = -\frac{a_{m-1}}{a_m} \quad (51)$$

error-free calculation of the defects (48) from the interpolation values (45).

## 6. The ECP-Rayleigh Quotient

Let be the eigenvalue equation ([3], p. 421)

$$\det \mathbf{F}(\lambda) = \det(\mathbf{E} - \lambda \mathbf{I}_m) = 0 \quad (52)$$

with the accompanying ECP matrix

$$\mathbf{E} = \text{Diag} \langle \sigma_j \rangle - \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} (d_1 \ d_2 \ \cdots \ d_m). \quad (53)$$

The Rayleigh quotient

$$R(\lambda) = \frac{\mathbf{y}^T(\lambda) \mathbf{E} \mathbf{x}(\lambda)}{\mathbf{y}^T(\lambda) \mathbf{I}_m \mathbf{x}(\lambda)}, \quad (54)$$

where

$$\mathbf{y}^T(\lambda) = \left( \frac{1}{\sigma_1 - \lambda} \quad \frac{1}{\sigma_2 - \lambda} \quad \cdots \quad \frac{1}{\sigma_m - \lambda} \right), \quad (55)$$

$$\mathbf{x}(\lambda) = \begin{pmatrix} \frac{d_1}{\sigma_1 - \lambda} \\ \frac{d_2}{\sigma_2 - \lambda} \\ \vdots \\ \frac{d_m}{\sigma_m - \lambda} \end{pmatrix} \quad (56)$$

can be reformulated by using the terms

$$S_1(\lambda) = \sum_{j=1}^m \frac{d_j}{\sigma_j - \lambda}, \quad (58)$$

$$S_2(\lambda) = \sum_{j=1}^m \frac{d_j}{(\sigma_j - \lambda)^2}, \quad (59)$$

$$S_\sigma(\lambda) = \sum_{j=1}^m \frac{d_j \sigma_j}{(\sigma_j - \lambda)^2} \quad (60)$$

in the form

$$R(\lambda) = \frac{S_\sigma(\lambda) - S_1^2(\lambda)}{S_2(\lambda)}. \quad (61)$$

Therefore, the following algorithm is defined

$$\Lambda_{j+1} = \Lambda_j + R(\Lambda_j); \quad j = 1, 2, \dots \quad (62)$$

which can be started by a main value  $H_k$ .

## 7. The Reduced Eigenvalue Equation

Among the eigenvalue equation (52) the reduced eigenvalue equation exists according to ([3], p. 346)

$$\tilde{f}(\lambda) = S_1(\lambda) - 1 = 0. \quad (63)$$

With the derivative

$$\tilde{f}'(\lambda) = S_1'(\lambda) - 0 = S_2(\lambda) \quad (64)$$

the Padé function

$$p_E(\lambda) = \frac{\tilde{f}(\lambda)}{-\tilde{f}'(\lambda)} = \frac{S_1(\lambda) - 1}{-S_2(\lambda)} \quad (65)$$

is obtained and therefore the algorithm

$$\Lambda_{j+1} = \Lambda_j + p_E(\Lambda_j); \quad j = 1, 2, \dots \quad (66)$$

It can be started by a main value  $H_k$ .

## 8. The Evolution

8a) An additional algorithm is introduced in ([3], p.44) which can be described as follows: replace the interpolation values in list (50) by the main values and prepare a new list; repeat this procedure as long as some or all defects go below a prescribed threshold. The main values of the final list can be used as start values of the algorithm in section 2.

## 9. Numerical Feasibility and Additional Aspects

9a) Evaluation of the multiplicity for the algorithm (18).

Execute the algorithm for  $\nu = 1$ ,  $\nu = 2$ , and so on, simultaneously. The convergence will be taken place exactly once. Therefore, the zeros and their multiplicity is determined.

The Taylor test with the characteristic polynomial (4)

$$\begin{aligned} \nu = 1: \quad & f(a) = 0 \\ & f'(a) \neq 0 \\ \nu = 2: \quad & f(a) = 0 \\ & f'(a) = 0 \\ & f''(a) \neq 0 \end{aligned} \quad (67)$$

and the same manner for  $\nu > 2$  can be used as control.

9b) The matrix (52) can be reformulated as

$$\mathbf{E} = \begin{pmatrix} H_1 & -d_2 & -d_3 & \cdots & -d_{m-1} & -d_m \\ -d_1 & H_2 & -d_3 & \cdots & -d_{m-1} & -d_m \\ -d_1 & -d_2 & H_3 & \cdots & -d_{m-1} & -d_m \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ -d_1 & -d_2 & -d_3 & \cdots & H_{m-1} & -d_m \\ -d_1 & -d_2 & -d_3 & \cdots & -d_{m-1} & H_m \end{pmatrix} \quad (68)$$

and therefore

$$\text{Tr } \mathbf{E} = \sum_{j=1}^m H_j = \sum_{j=1}^m \lambda_j = -\frac{a_{m-1}}{a_m} \quad (69)$$

such that eq. (51) is proved.

9c) Gershgorin's circle theorems by means of the matrix

$$\mathbf{F}(\lambda) = \mathbf{E} - \lambda \mathbf{I}_m. \quad (70)$$

Let be a circle with the center point

$$H_k = U_k + V_k \cdot i \quad (71)$$

and the radius

$$r_k = (n - 1) \cdot |d_k|. \quad (72)$$

If the circle is separated from the remaining  $n - 1$  circles then we have to distinguish two cases

9c1) The main value  $H_k$  is real. Then also the included eigenvalue  $\lambda_k$  is real and we have

$$-r_k + H_k < \lambda_k < H_k + r_k. \quad (73)$$

9c2) For a complex eigenvalue

$$\lambda_k = u_k + v_k \cdot i \quad (74)$$

we have the enclosures

$$-r_k + U_k < u_k < U_k + r_k \quad (75)$$

and

$$-r_k + V_k < v_k < V_k + r_k. \quad (76)$$

In the case of multiple eigenvalues or eigenvalue clusters we have simultaneous enclosures; cf. ([3], p.52).

9d) Order reduction

9d1) Scalar Polynomial. Separated a zero using Horner's scheme.

9d2) Matrix polynomial (2). Separated a cluster of  $n$  eigenvalues en bloc [5].

## 10. Numerical Examples

### Example 1:

Following section 3 an exploration is performed by means of the Padé function (6). The polynomial

$$f(\lambda) = 4 + 12\lambda + 9\lambda^2 - 4\lambda^3 - 6\lambda^4 + 0 \cdot \lambda^5 + \lambda^6 \quad (77)$$

is assumed with the zeros

$$\lambda_1 = \lambda_2 = 2; \lambda_3 = \lambda_4 = \lambda_5 = \lambda_6 = -1. \quad (78)$$

An exploration with  $\delta = 0.3$  results in the pairs of values

$\Lambda$	$p(\Lambda)$	
$3.000000000000000e-01$	$-5.261904761904761e-01$	}
$6.000000000000000e-01$	$-9.333333333333332e-01$	
$9.000000000000000e-01$	$-3.483333333333336e+00$	
$1.200000000000000e+00$	$+1.466666666666667e+00$	
$1.500000000000000e+00$	$+4.166666666666667e-01$	
$\lambda_1 = 1.800000000000000e+00$	$+1.166666666666663e-01$	}
$\lambda_2 = 2.100000000000000e+00$	$-4.696969696969665e-02$	

(79)

change of sign

It follows the regula falsi (21) with a rounding to

$$\lambda_3 = 2.01389 \quad (80)$$



and therefore with (18)

$j$	$\Lambda_j$	$p_1(\Lambda_j)$
1	2.0138900000000000e+00	-3.428770022238466e-03
2	2.006984834339914e+00	-1.735089829011837e-03
3	2.003502535366870e+00	-8.728302942998172e-04
4	2.001753817659295e+00	-4.377505086146021e-04
5	2.000877548907494e+00	-2.192108705613412e-04

(81)

There is no convergence for  $\nu = 1$ .

$j$	$\Lambda_j$	$p_2(\Lambda_j)$
1	2.0138900000000000e+00	-6.734450893113345e-03
2	2.000327556690868e+00	-1.636577281729843e-04
3	2.000000187627338e+00	-9.381362945114195e-08
4	2.0000000000000062e+00	-3.090806074727201e-14
5	2.0000000000000000e+00	0

(82)

It converges for  $\nu = 2$  and therefore we have

$$\lambda_1 = \lambda_2 = 2. \quad (83)$$

If  $\lambda$  is replaced by  $-\lambda$  we obtain the co-polynomial

$$f(-\lambda) = 4 - 12\lambda + 9\lambda^2 + 4\lambda^3 - 6\lambda^4 - 0 \cdot \lambda^5 + \lambda^6. \quad (84)$$

An exploration results in

$\Lambda$	$p(\Lambda)$
3.0000000000000000e-01	-2.064102564102564e-01
6.0000000000000000e-01	-1.0833333333333336e-01
9.0000000000000000e-01	-2.543859649125041e-02
1.2000000000000000e+00	+4.848484848484354e-02
1.5000000000000000e+00	+1.166666666666667e-01

(85)

It follows the regula falsi (21) with a rounding to

$$\lambda_3 = 1.00324. \quad (86)$$

With algorithm (18) we obtain no convergence for  $\nu = 1$ ,  $\nu = 2$ ,  $\nu = 3$  but for  $\nu = 4$

$j$	$\Lambda_j$	$p_4(\Lambda_j)$
1	1.0032400000000000e+00	-3.191729984318953e-03
2	1.000037928810532e+00	-3.792209825976858e-05
3	1.000000005273932e+00	-5.273931558410046e-09
4	1.0000000000000000e+00	-2.343804163097548e-16

(87)

We have four times +1 of the co-polynomial and therefore

$$\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = -1. \quad (88)$$

With MATLAB the following zeros are calculated

$$\begin{aligned} \tilde{\lambda}_1 &= +2.0000000000000001e+00 + 7.152216756864169e-09i \\ \tilde{\lambda}_2 &= +2.0000000000000001e+00 - 7.152216756864169e-09i \\ \tilde{\lambda}_3 &= -1.000143391292847e+00 \\ \tilde{\lambda}_4 &= -9.999999991419022e-01 + 1.433904397109860e-04i \\ \tilde{\lambda}_5 &= -9.999999991419022e-01 - 1.433904397109860e-04i \\ \tilde{\lambda}_6 &= -9.998566104233441e-01 \end{aligned} \quad (89)$$

### Example 2:

Let us consider a diagonal dominant matrix

$$\mathbf{F}(\lambda) = \begin{pmatrix} 5 + 2\lambda + 3\lambda^2 & -1 & 0 & 0 & 0 \\ -1 & 9 + 3\lambda + \lambda^2 & -3 & -2 & 0 \\ 0 & -3 & 6 + \lambda^2 & -2 & 0 \\ 0 & -2 & 2 & 12 + \lambda + \lambda^2 & -5 - \lambda \\ 0 & 0 & 0 & -5 - \lambda & 8 + 4\lambda + 4\lambda^2 \end{pmatrix} \quad (90)$$

with

$$\det \mathbf{F}(\lambda) = f(\lambda) = 1221 + 19366\lambda + 33492\lambda^2 + 28079\lambda^3 + 23637\lambda^4 + 11574\lambda^5 + 5699\lambda^6 + 1631\lambda^7 + 489\lambda^8 + 68\lambda^9 + 12\lambda^{10} \quad (91)$$

The quadratic equations (29)

$$\begin{aligned} 5 + 2\lambda + 3\lambda^2 &= 0 \\ 9 + 3\lambda + \lambda^2 &= 0 \\ 6 + \lambda^2 &= 0 \\ 12 + \lambda + \lambda^2 &= 0 \\ 8 + 4\lambda + 4\lambda^2 &= 0 \end{aligned} \quad (92)$$

have the zeros

$$\begin{aligned} &-3.333333333333334e - 01 + 1.247219128924647e + 00i \\ &-3.333333333333334e - 01 - 1.247219128924647e + 00i \\ &-1.500000000000000e + 00 + 2.598076211353316e + 00i \\ &-1.500000000000000e + 00 - 2.598076211353316e + 00i \\ &0 + 2.449489742783178e + 00i \\ &0 - 2.449489742783178e + 00i \\ &-4.999999999999998e - 01 + 3.427827300200522e + 00i \\ &-4.999999999999998e - 01 - 3.427827300200522e + 00i \\ &-5.000000000000000e - 01 + 1.322875655532295e + 00i \\ &-5.000000000000000e - 01 - 1.322875655532295e + 00i \end{aligned} \quad (93)$$

We choose the third row as from (93) as starting value and obtain the following results.

a) Algorithm (11)

$j$	$\Lambda_j$	$p_4(\Lambda_j)$
1	$-1.500000000000000e + 00 + 2.598076211353316e + 00i$	$+2.032105570683291e - 01 - 7.395719545396148e - 02i$
2	$-1.296789442931671e + 00 + 2.524119015899355e + 00i$	$+1.773110756693907e - 01 - 2.275434281511629e - 02i$
3	$-1.119478367262280e + 00 + 2.501364673084238e + 00i$	$+1.320427191450034e - 01 + 5.627450679466858e - 02i$
4	$-9.874356481172767e - 01 + 2.557639179878907e + 00i$	$-3.007105982831948e - 02 + 8.920521695652191e - 02i$
5	$-1.017506707945596e + 00 + 2.646844396835429e + 00i$	$+7.359853863202630e - 04 - 2.137437648438716e - 02i$
6	$-1.016770722559276e + 00 + 2.625470020351042e + 00i$	$-9.748769601593532e - 04 - 1.081681076639398e - 03i$
7	$-1.017745599519435e + 00 + 2.624388339274403e + 00i$	$-5.137187053597404e - 06 + 4.029601688399558e - 06i$
8	$-1.017750736706489e + 00 + 2.624392368876091e + 00i$	$+1.136121179599617e - 10 - 6.578228494562704e - 11i$
9	$-1.017750736592877e + 00 + 2.624392368810308e + 00i$	$+2.009798662754188e - 15 - 1.502768992132371e - 15i$

(94)

b) Algorithm (18)

$j$	$\Lambda_j$	$p_4(\Lambda_j)$
1	$-1.500000000000000e+00 + 2.598076211353316e+00i$	$-5.590550503850626e-02 - 5.178554874365029e-02i$
2	$-1.281598940159485e+00 + 2.530507771744555e+00i$	$-3.206896765146202e-02 - 5.435962387848091e-02i$
3	$-1.102941934511756e+00 + 2.519024236220835e+00i$	$+4.957773050493467e-03 - 4.558545263614854e-02i$
4	$-9.935792103013821e-01 + 2.581791094008818e+00i$	$+2.032174795110959e-02 + 2.087166993263317e-03i$
5	$-1.019159105737506e+00 + 2.632183836150752e+00i$	$-2.713265716958361e-03 + 4.879297147422435e-04i$
6	$-1.017678176984104e+00 + 2.624544743975652e+00i$	$-4.116398540465768e-05 + 4.357842452009323e-05i$
7	$-1.017745599519435e+00 + 2.624388339274403e+00i$	$-5.137187053597404e-06 + 4.029601688399558e-06i$
8	$-1.017750658819505e+00 + 2.624392358442496e+00i$	$+1.342417115296137e-08 + 2.442886475494064e-08i$
9	$-1.017750736592890e+00 + 2.624392368810295e+00i$	$+2.991635783682046e-15 - 6.560209464970359e-15i$

(95)

Both algorithms converge quadratic and deliver almost identical results. In the same manner the remaining nine zeros will be calculated in parallel and independent from each other.

### Example 3:

Evolution following section 8.

The polynomial denoted after Wilkinson

$$f(\lambda) = (1 - \lambda)(2 - \lambda) \cdots (9 - \lambda)(10 - \lambda) \quad (96)$$

or in a decomposed form

$$f(\lambda) = 3828800 - 10628640\lambda + 12753576\lambda^2 - 8409500\lambda^3 + 3416930\lambda^4 - 902055\lambda^5 + \quad (97)$$

$$+157773\lambda^6 - 18150\lambda^7 + 1320\lambda^8 - 55\lambda^9 + \lambda^{10} \quad (98)$$

has the zeros  $1, 2, \dots, 10$ .

Calculated zeros with MATLAB

$$\begin{aligned} \tilde{\lambda}_1 &= 1.000000000032865e+01 \\ \tilde{\lambda}_2 &= 8.999999998364443e+00 \\ \tilde{\lambda}_3 &= 8.000000003420013e+00 \\ \tilde{\lambda}_4 &= 6.999999996085851e+00 \\ \tilde{\lambda}_5 &= 6.000000002669752e+00 \\ \tilde{\lambda}_6 &= 4.999999998898655e+00 \\ \tilde{\lambda}_7 &= 4.000000000263102e+00 \\ \tilde{\lambda}_8 &= 2.99999999968169e+00 \\ \tilde{\lambda}_9 &= 2.00000000001345e+00 \\ \tilde{\lambda}_{10} &= 1.000000000000000e+00 \end{aligned} \quad (99)$$

We use these values as interpolation values and obtain the following list

$$L_{10} = \begin{pmatrix} \begin{array}{ccc} \text{Interpolation Values} & \text{Defects} & \text{Main Values} \end{array} \\ \begin{array}{l} 1.000000000032865e+01 \\ 8.999999998364443e+00 \\ 8.000000003420013e+00 \\ 6.999999996085851e+00 \\ 6.000000002669752e+00 \\ 4.999999998898655e+00 \\ 4.000000000263102e+00 \\ 2.99999999968169e+00 \\ 2.00000000001345e+00 \\ 1.000000000000000e+00 \end{array} & \begin{array}{l} +3.727125322099494e-10 \\ -1.720094133611112e-09 \\ +3.167697847832428e-09 \\ -4.044785689387324e-09 \\ +2.348194056725277e-09 \\ -1.197945997413012e-09 \\ +2.777798930869391e-10 \\ -3.340011018696152e-11 \\ +1.212659602373197e-12 \\ +0.000000000000000e+00 \end{array} & \begin{array}{l} 9.99999999955941e+00 \\ 9.00000000084537e+00 \\ 8.000000000252316e+00 \\ 7.000000000130637e+00 \\ 6.000000000321559e+00 \\ 5.00000000096601e+00 \\ 3.99999999985322e+00 \\ 3.00000000001569e+00 \\ 2.00000000000132e+00 \\ 1.000000000000000e+00 \end{array} \end{pmatrix} \quad (100)$$

It follows two evolutions

$$L_{10} = \begin{pmatrix} \begin{array}{ccc} \text{Interpolation Values} & \text{Defects} & \text{Main Values} \\ 9.99999999955941e+00 & -5.567454977060759e-11 & 1.000000000001162e+01 \\ 9.000000000084537e+00 & -3.564064318273009e-11 & 9.000000000120178e+00 \\ 8.000000000252316e+00 & -8.075158037691484e-11 & 8.000000000333067e+00 \\ 7.000000000130637e+00 & +2.082540757136542e-10 & 6.99999999922383e+00 \\ 6.000000000321559e+00 & +2.186021042651543e-10 & 6.000000000102957e+00 \\ 5.000000000096601e+00 & +6.758556180612028e-11 & 5.00000000020015e+00 \\ 3.99999999985322e+00 & -1.347399557668777e-11 & 3.99999999998797e+00 \\ 3.00000000001569e+00 & +4.296279733101633e-12 & 2.99999999997273e+00 \\ 2.00000000000132e+00 & +6.929483440822838e-14 & 2.000000000000063e+00 \\ 1.000000000000000e+00 & +0.000000000000000e+00 & 1.000000000000000e+00 \end{array} \end{pmatrix} \quad (101)$$

$$L_{10} = \begin{pmatrix} \begin{array}{ccc} \text{Interpolation Values} & \text{Defects} & \text{Main Values} \\ 1.000000000001162e+01 & +1.774461056701600e-11 & 9.99999999993872e+00 \\ 9.000000000020178e+00 & -7.047284661419908e-11 & 9.000000000190651e+00 \\ 8.000000000333067e+00 & -7.797978696197630e-11 & 8.000000000411047e+00 \\ 6.99999999922383e+00 & -3.380356010780448e-10 & 7.000000000260418e+00 \\ 6.000000000102957e+00 & +1.409918897505596e-10 & 5.99999999961965e+00 \\ 5.00000000029015e+00 & +7.793359042933322e-11 & 4.99999999951082e+00 \\ 3.9999999998797e+00 & -5.928558055009481e-12 & 4.000000000004725e+00 \\ 2.99999999997273e+00 & -2.910383045325759e-12 & 3.000000000000184e+00 \\ 2.000000000000063e+00 & +4.619655627634508e-14 & 2.000000000000016e+00 \\ 1.000000000000000e+00 & +0.000000000000000e+00 & 1.000000000000000e+00 \end{array} \end{pmatrix} \quad (102)$$

Using the sum control (51) we have ( $a_{10} = 1$ )

$$\text{Desired value: } \sum_{j=1}^{10} H_j = -\frac{-55}{a_{10}} = 55. \quad \text{Actual value: } 49.466 \quad (103)$$

#### Example 4:

Correction of multiple eigenvalues after (18).

Let be a matrix pencil

$$\mathbf{F}(\lambda) = \mathbf{A} - \lambda \mathbf{B} \quad (104)$$

with

$$\mathbf{A} = \begin{pmatrix} -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 \end{pmatrix}; \mathbf{B} = \mathbf{I}_5. \quad (105)$$

Its eigenvalues are

$$\lambda_1 = -2; \lambda_2 = \lambda_3 = -1; \lambda_4 = \lambda_5 = 1. \quad (106)$$

MATLAB calculates the following approximated zeros

$$\begin{aligned} \tilde{\lambda}_1 &= -1.999999999999996e+00 \\ \tilde{\lambda}_2 &= +1.000000000000000e+00 + 7.768125062636118e-09i \\ \tilde{\lambda}_3 &= +1.000000000000000e+00 - 7.768125062636118e-09i \\ \tilde{\lambda}_4 &= -1.000000009896685e+00 \\ \tilde{\lambda}_5 &= -9.999999901033162e-01 \end{aligned} \quad (107)$$

$$(108)$$

It follows the corrections after (18). We start with  $\tilde{\lambda}_5$ .

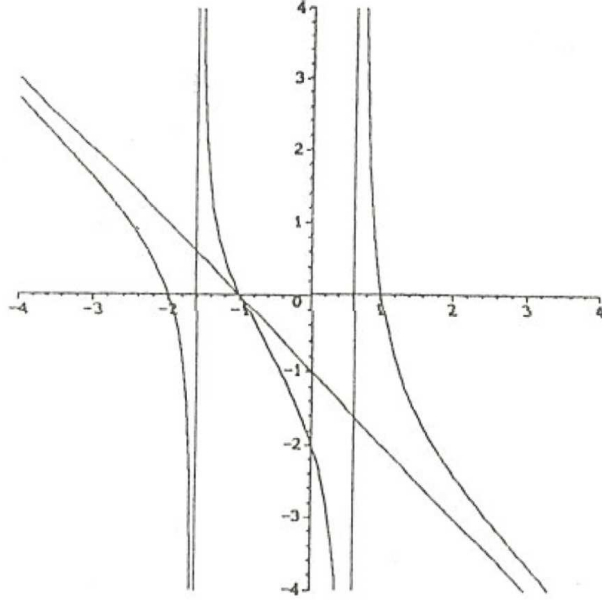


Figure 2: The Padé function of the matrix pencil of example 4

$j$	$\Lambda_j$	$p_1(\Lambda_j)$	
1	$-9.999999901033162e - 01$	$-1.121813169708485e - 08$	(109)
2	$-1.000000001321448e + 00$	0	

No convergence for  $\nu = 1$ .

$j$	$\Lambda_j$	$p_2(\Lambda_j)$	
1	$-9.999999901033162e - 01$	$-9.896683722532278e - 09$	(110)
2	$-9.999999999999999e - 01$	$-1.665334536937735e - 16$	

Convergence for  $\nu = 2$ . Therefore, we have  $\lambda_4 = \lambda_5 = 1$ .

We start with  $\tilde{\lambda}_1$ :

$j$	$\Lambda_j$	$p_1(\Lambda_j)$	
1	$-1.999999999999999e + 00$	$-3.552713678800562e - 15$	(111)
2	$-2.000000000000000e + 00$	0	

Therefore we have  $\lambda_1 = -2$ .

Now, we start with  $\tilde{\lambda}_2$ :

$j$	$\Lambda_j$	$p_1(\Lambda_j)$	
1	$1.000000000000000e + 00 + 7.768125062636118e - 09i$	$-1.491714852512857e - 16 - 4.764011116681661e - 09i$	(112)
2	$1.000000000000000e + 00 + 3.004113945954457e - 09i$	$-2.065119601239563e - 16 - 1.334040524532894e - 23i$	

No convergence for  $\nu = 1$ .

$j$	$\Lambda_j$	$p_2(\Lambda_j)$	
1	$1.000000000000000e + 00 + 7.768125062636118e - 09i$	$-5.744419753425677e - 16 - 7.768125062636107e - 09i$	(113)
2	$1.000000000000000e + 00 + 3.004113945954457e - 09i$	$+1.295260195396017e - 16 - 1.158052857574239e - 23i$	

Convergence for  $\nu = 2$ .

Therefore, we have  $\lambda_2 = \lambda_3 = -1$ .

**Example 5:**

Multiple complex zeros: The polynomial

$$f(\lambda) = (1 + \lambda + \lambda^2)^3 \cdot (1 + \lambda^2)^2 \cdot 6 \quad (114)$$

can be decomposed into

$$f(\lambda) = 6 + 18\lambda + 48\lambda^2 + 78\lambda^3 + 114\lambda^6 + 78\lambda^7 + 48\lambda^8 + 18\lambda^9 + 6\lambda^{10}. \quad (115)$$

The zeros are

$$\begin{aligned} \lambda_1 &= 0 + i \\ \lambda_2 &= 0 - i \\ \lambda_3 &= 0 + i \\ \lambda_4 &= 0 - i \\ \lambda_5 &= -0,5 + \sqrt{0,75}i \\ \lambda_6 &= -0,5 - \sqrt{0,75}i \\ \lambda_7 &= -0,5 + \sqrt{0,75}i \\ \lambda_8 &= -0,5 - \sqrt{0,75}i \\ \lambda_9 &= -0,5 + \sqrt{0,75}i \\ \lambda_{10} &= -0,5 - \sqrt{0,75}i \end{aligned} \quad (116)$$

with

$$\sqrt{0,75} = 8.660254037844386e - 01. \quad (117)$$

MATLAB calculates the following approximated zeros

$$\begin{aligned} \tilde{\lambda}_1 &= +2.103940549558203e - 08 + 1.000000028920264e + 00i \\ \tilde{\lambda}_2 &= +2.103940549558203e - 08 - 1.000000028920264e + 00i \\ \tilde{\lambda}_3 &= -2.103939766850971e - 08 + 9.999999710797240e - 01i \\ \tilde{\lambda}_4 &= -2.103939766850971e - 08 - 9.999999710797240e - 01i \\ \tilde{\lambda}_5 &= -5.000094136551562e - 01 + 8.660276783463672e - 01i \\ \tilde{\lambda}_6 &= -5.000094136551562e - 01 - 8.660276783463672e - 01i \\ \tilde{\lambda}_7 &= -4.999933232335635e - 01 + 8.660324192348879e - 01i \\ \tilde{\lambda}_8 &= -4.999933232335635e - 01 - 8.660324192348879e - 01i \\ \tilde{\lambda}_9 &= -4.999972631112927e - 01 + 8.660161137720683e - 01i \\ \tilde{\lambda}_{10} &= -4.999972631112927e - 01 - 8.660161137720683e - 01i \end{aligned} \quad (118)$$

Correction of  $\tilde{\lambda}_5$  after (18).

$j$	$\Lambda_j$	$p_1(\Lambda_j)$
1	$-5.000094136551562e - 01 + 8.660276783463672e - 01i$	$+2.590606103860521e - 06 - 4.667337884496285e - 07i$
2	$-5.000068230490523e - 01 + 8.660272116125788e - 01i$	$+5.061285460948595e - 06 - 9.728311147477021e - 07i$
3	$-5.000017617635913e - 01 + 8.660262387814640e - 01i$	$-3.494617894702583e - 05 + 1.077378144656732e - 05i$
4	$-5.000367079425384e - 01 + 8.660370125629105e - 01i$	$+1.215888288037037e - 05 - 3.935532380990325e - 06i$
5	$-5.000245490596580e - 01 + 8.660330770305296e - 01i$	$+2.065119601239563e - 06 - 2.552780638469371e - 06i$

(119)

No convergence for  $\nu = 1$ .

$j$	$\Lambda_j$	$p_2(\Lambda_j)$
1	$-5.000094136551562e - 01 + 8.660276783463672e - 01i$	$4.706686932467359e - 06 - 1.137278373809355e - 06i$
2	$-5.000047069682237e - 01 + 8.660265410679934e - 01i$	$2.353445844274929e - 06 - 5.686590462229913e - 07i$
3	$-5.000023535223794e - 01 + 8.660259724089472e - 01i$	$1.176760802740233e - 06 - 2.842232846214180e - 07i$
4	$-5.000011767615767e - 01 + 8.660256881856626e - 01i$	$5.883348476664755e - 07 - 1.420315348699738e - 07i$
5	$-5.000005884267291e - 01 + 8.660255461541277e - 01i$	$2.940921445040524e - 07 - 7.102102632529488e - 08i$

(120)

No convergence for  $\nu = 2$ .

$j$	$\Lambda_j$	$p_3(\Lambda_j)$
1	$-5.000000008552082e - 01 + 8.660254038125426e - 01i$	$+8.552084998311037e - 10 - 2.810326544530710e - 11i$
2	$-4.99999999999997e - 01 + 8.660254037844393e - 09i$	$-1.553244317250170e - 15 - 2.433903049193858e - 15i$

(121)

Convergence for  $\nu = 3$  such that we have a zero  $\lambda_5$  of (116) with the multiplicity 3. The polynomial (115) is (accidental) hermitian but of even order  $m = 10$  and therefore  $-1$  is no zero.

**Example 6:**

The reduced eigenvalues equation (66) with

$$p_E(\lambda) = \frac{S_1(\lambda) - 1}{-S_2(\lambda)} \quad (122)$$

Wilkinson polynomial (96):

$$L_{10} = \begin{pmatrix} \begin{array}{ccc} \text{Interpolation Values} & \text{Defects} & \text{Main Values} \end{array} \\ \begin{array}{ccc} 1.000100000001162e + 01 & 9.988178397837826e - 05 & 1.000000118216022e + 00 \\ 2.000200000020178e + 00 & 1.997514392797159e - 04 & 2.000000248560720e + 00 \\ 3.000300000333067e + 00 & 2.996318462367703e - 04 & 3.000000368153763e + 00 \\ 4.000400000000000e + 00 & 3.995415371807089e - 04 & 4.000000458462819e + 00 \\ 5.000500000000000e + 00 & 4.995001173401799e - 04 & 5.000000499882660e + 00 \\ 6.000599999999999e + 00 & 5.995318981743213e - 04 & 6.000000468101825e + 00 \\ 7.000699999999999e + 00 & 6.996716090711247e - 04 & 7.000000328390928e + 00 \\ 8.000800000000000e + 00 & 7.999787558876377e - 04 & 8.000000021244112e + 00 \\ 9.000900000000000e + 00 & 9.005807841077982e - 04 & 8.999999419215891e + 00 \\ 1.000100000000000e + 01 & 1.001930111657591e - 03 & 9.999998069888342e + 00 \end{array} \end{pmatrix} \quad (123)$$

We start with the main value  $H_3$  an obtain

$j$	$\Lambda_j$	$p_2(\Lambda_j)$
1	$3.000000368155010e + 00$	$-3.677030854107595e - 07$
2	$3.000000000451924e + 00$	$-4.517926727218366e - 10$
3	$3.000000000000131e + 00$	$-5.277591961897212e - 16$

(124)

In comparison with the algorithm (11)

$j$	$\Lambda_j$	$p_2(\Lambda_j)$
1	$3.000000368155010e + 00$	$-3.681554823412160e - 07$
2	$2.99999999999527e + 00$	$+2.448417482860382e - 12$
3	$3.000000000001975e + 00$	$-8.777345693336323e - 13$

(125)

**Example 7:**

Singular leading matrix.

We assume a polynomial matrix

$$\mathbf{F}(\lambda) = \mathbf{A}_0 + \mathbf{A}_1\lambda + \mathbf{A}_2\lambda^2 + \mathbf{A}_3\lambda^3 + \mathbf{A}_4\lambda^4 \quad (126)$$

with the coefficient matrices

$$\begin{aligned} \mathbf{A}_0 &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \\ \mathbf{A}_1 &= \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}, \\ \mathbf{A}_2 &= \begin{pmatrix} 2 & 1 \\ 0 & 1 \end{pmatrix}, \\ \mathbf{A}_3 &= \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}, \\ \mathbf{A}_4 &= \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \end{aligned} \quad (127)$$

where the leading matrix  $\mathbf{A}_4$  is singular such that we have fewer than  $m = \rho \cdot n$  eigenvalues.

We have to distinguish two approaches:

a) using the matrix

$$\mathbf{F}(\lambda) = \begin{pmatrix} 1 + \lambda + 2\lambda^2 & \lambda + 2\lambda^2 + \lambda^4 \\ \lambda & 1 + \lambda + 2\lambda^2 \end{pmatrix} \quad (128)$$

using the characteristic polynomial

$$\det \mathbf{F}(\lambda) = f(\lambda) = 1 + 2\lambda + 3\lambda^2 + 2\lambda^3 + 2\lambda^4 - \lambda^5 \quad (129)$$

with the degree 5; therefore, we have only 5 zeros and accordingly 5 eigenvalues.

We start with the exploration using the Padé function (6) and choose  $\delta = 0.1$ :

$\Lambda_j$	$p(\Lambda_j)$	
0.0	-2.064102564102564e-01	}
0.1	-2.064102564102564e-01	
0.2	-2.064102564102564e-01	
0.3	-2.064102564102564e-01	
0.4	-2.064102564102564e-01	
0.5	-2.064102564102564e-01	
0.6	-2.064102564102564e-01	
0.7	-2.064102564102564e-01	
0.8	-2.064102564102564e-01	
0.9	-2.064102564102564e-01	
1.0	-2.064102564102564e-01	
1.1	-2.064102564102564e-01	
1.2	-2.064102564102564e-01	
1.3	-2.064102564102564e-01	
1.4	-2.064102564102564e-01	
1.5	-2.064102564102564e-01	
1.6	-2.064102564102564e-01	
1.7	-2.064102564102564e-01	
1.8	-2.064102564102564e-01	
1.9	-2.064102564102564e-01	
2.0	-2.064102564102564e-01	
2.1	-2.064102564102564e-01	
2.2	-2.064102564102564e-01	
2.3	-2.064102564102564e-01	
2.4	-2.064102564102564e-01	
2.5	-2.064102564102564e-01	
2.6	-2.064102564102564e-01	
2.7	-2.064102564102564e-01	
2.8	-2.064102564102564e-01	
2.9	-2.064102564102564e-01	
$\lambda_1 =$ 3.0	-2.064102564102564e-01	} change of sign
$\lambda_2 =$ 3.1	-2.064102564102564e-01	

and hence with the regula falsi

$$\lambda_3 = 3.05965871206409e+00 \quad (131)$$

and furthermore after (26) with  $\sigma = 5$

$$\lambda_4 = 3.056811621817845e+00. \quad (132)$$

It follows the Padé algorithm (11)



$j$	$\Lambda_j$	$p(\Lambda_j)$
1	$3.056811621817845e + 00$	$-2.231403025508382e - 06$
2	$3.056809390414819e + 00$	$-5.754993999073329e - 12$
3	$3.056809390409065e + 00$	$-7.589857143243228e - 17$

(133)

a) MATLAB calculates the eigenvalues for the matrix (128)

$$\begin{aligned}
 \tilde{\lambda}_1 &= +3.056809390409061e + 00 \\
 \tilde{\lambda}_2 &= -2.103940549558203e - 08 - 1.000000028920264e + 00i \\
 \tilde{\lambda}_3 &= -2.103939766850971e - 08 + 9.999999710797240e - 01i \\
 \tilde{\lambda}_4 &= -2.103939766850971e - 08 - 9.999999710797240e - 01i \\
 \tilde{\lambda}_5 &= -5.000094136551562e - 01 + 8.660276783463672e - 01i
 \end{aligned}$$
(134)

b) MATLAB calculates the zeros for the polynomial (129)

$$\begin{aligned}
 \tilde{\lambda}_1 &= +3.056809390409070e + 00 \\
 \tilde{\lambda}_2 &= -2.103940549558203e - 08 - 1.000000028920264e + 00i \\
 \tilde{\lambda}_3 &= -2.103939766850971e - 08 + 9.999999710797240e - 01i \\
 \tilde{\lambda}_4 &= -2.103939766850971e - 08 - 9.999999710797240e - 01i \\
 \tilde{\lambda}_5 &= -5.000094136551562e - 01 + 8.660276783463672e - 01i
 \end{aligned}$$
(135)

Both MATLAB results as well as  $\Lambda_3$  in (141) are comparable with respect to the accuracy

**Example 8:**

$$f(\lambda) = (\lambda - 1)(\lambda - 2)(\lambda - 3)(\lambda - 4)(\lambda - 5) \cdot 3$$
(136)

or in a decomposed form

$$f(\lambda) = -360 + 822\lambda - 675\lambda^2 + 255\lambda^3 - 45\lambda^4 + 3\lambda^5.$$
(137)

The exploration with  $\delta = 0.3$  delivers the pairs of values

	$\Lambda_j$	$p(\Lambda_j)$
	0.0	$+4.379562043795621e - 01$
	0.3	$+3.484061594869381e - 01$
	0.6	$+2.408279034112688e - 01$
$\lambda_1 =$	0.9	$+8.366965417990657e - 02$
$\lambda_2 =$	1.2	$-3.884787018255549e - 01$

$\left. \begin{array}{l} \} \\ \} \\ \} \end{array} \right\} \text{change of sign}$

(138)

With the regula falsi algorithm the following value can be calculated

$$\lambda_3 = 9.531631550437540e - 01.$$
(139)

Now, we use Halley's algorithm after (14).

$j$	$\Lambda_j$	$h(\Lambda_j)$
1	$9.000000000000000e - 01$	$+1.180808950230746e - 01$
2	$1.018080895023075e + 00$	$-1.738293306349123e - 02$
3	$1.000697961959583e + 00$	$-6.969460936557861e - 04$
4	$1.000001015865928e + 00$	$-1.015863777488736e - 06$
5	$1.000000000002150e + 00$	$-2.150576013567233e - 12$
6	$9.999999999999994e - 01$	$-7.894919286223337e - 16$
7	$1.000000000000000e + 00$	$+0.000000000000000e + 00$

(140)

It is known from the theory [6]: the convergence of Halley's algorithm is cubic for simple zeros and quadratic for multiple zeros. However, at least in this example a cubic convergence cannot be observed.

In comparison: the accelerated regula falsi following (139) leads in five steps to the nearly exact solution

$j$	$\lambda_j$	$p(\lambda_j)$	(141)
4	$9.985736255069474e - 01$	$+1.422152519655597e - 03$	
5	$9.999780098758768e - 01$	$+2.198911675590514e - 05$	
6	$1.000000002011310e + 00$	$-2.011309654213950e - 09$	
7	$1.000000000000000e + 00$	$+0.000000000000000e + 00$	

**Acknowledgement:** I would like to thank M. Sc. Jonas Denißen, Berlin, and Prof. Dr.-Ing. Wolfgang Mathis, Leibniz University of Hannover, for the successful cooperation.

### References

- [1] Carstensen, C.; E. Stein (1987): Analysis and Calculation of Falk's ECP-transformation and familiar problems (in German). Intern. Series Numerical Mathem., vol. 83, pp. 47-61
- [2] Carstensen, C.; E. Stein (1989): About Falk's ECP-transformation and generalizations (in German). ZAMM, vol. 69, no. 11, pp. 375-391
- [3] Falk, S. (2004): The eigenvalue algorithm ECP for polynomial matrices (in German). Abhandl. Braunsch. Wiss. Gesellsch., vol. 54
- [4] Falk, S. (2011): The accelerated Ritz iteration for polynomials and polynomial matrices (in German). Abhandl. Braunsch. Wiss. Gesellsch., vol. 58
- [5] Falk, S. (1994): The reduction method for polynomial matrices (in German). ZAMM, vol. 74, no. 1, pp. 3-15
- [6] Mathis, W.; T. Thiessen; S. Falk (2010): Iteration methods for zero problems (in German). Institute of Theoretical Electrical Engineering, Leibniz University of Hannover
- [7] Shaw, M.; J.F. Traub (1974): In the number of multiplications for the evaluation of a polynomial and some of its derivations. J. Assoc. Comp. mech., vol. 21, pp. 61-166
- [8] Stewart, G. W. (1968): Some topics in numerical analysis. I Lehmer's method for finding the zeros of a polynomial. Techn. report, Oak Ridge
- [9] Zurmühl, R.; S. Falk (1992): Matrices and its applications 1 (in German). 6. edition, Springer-Verlag, Berlin
- [10] Zurmühl, R.; S. Falk (1986): Matrices and its applications 2 (in German). 5. edition, Springer-Verlag, Berlin